CS 4320/5320 Homework 1 Solutions

Spring 2017

March 25, 2017

1 SQL Queries

(a)
```
select area, count(*) from Papers group by area;
```

(b)
```
select name1, name2 from (
select R1.name as name1, R2.name as name2, count(*) as papers from
    Researchers as R1,
    Researchers as R2,
    Authorship as A1,
    Authorship as A2
    where R1.rid = A1.rid and R2.rid = A2.rid and
           A1.pid = A2.pid and R1.rid < R2.rid
    group by R1.rid, R2.rid) as T where T.papers in (
        select MAX(U.papers) from (
            select R1.name as name1, R2.name as name2, count(*) as papers from
                Researchers as R1,
                Researchers as R2,
                Authorship as A1,
                Authorship as A2
                where R1.rid = A1.rid and R2.rid = A2.rid and
                        A1.pid = A2.pid and R1.rid < R2.rid
                group by R1.rid, R2.rid) as U);
```

(c)
```
select distinct R.name, T.area from
        Researchers as R,
        (select distinct P.area from Papers as P) as T
    where not exists
        (select distinct R0.rid from Researchers as R0, Authorship as A, Papers as P
        where R.rid = R0.rid and R0.rid = A.rid and P.pid = A.pid and P.area = T.area);
```

(d)
```
select IntersectionInfo.id1, IntersectionInfo.id2,
    IntersectionInfo.size / UnionInfo.size as Jaccard
from
    (select A1.rid as id1,
           A2.rid as id2,
           COUNT(distinct A1.pid) as size
    from Authorship A1, Authorship A2
    where A1.rid < A2.rid and A1.pid = A2.pid
    group by A1.rid, A2.rid
    union
    select A1.rid as id1,
            A2.rid as id2,
            0 as size
    from Authorship A1, Authorship A2
    where A1.rid < A2.rid and NOT EXISTS
        (select * from Authorship P3, Authorship P4
        where A1.rid = P3.rid and A2.rid = P4.rid and P3.pid = P4.pid))
as IntersectionInfo,

    (select Temp.id1, Temp.id2, COUNT(*) as size
    from
        (select A1.rid as id1,
                A2.rid as id2,
                A1.pid
        from Authorship A1, Authorship A2
        where A1.rid < A2.rid
        union
        select A1.rid, A2.rid, A2.pid
        from Authorship A1, Authorship A2
        where A1.rid < A2.rid)
    as Temp
    group by Temp.id1, Temp.id2)
as UnionInfo

where IntersectionInfo.id1 = UnionInfo.id1 and
    IntersectionInfo.id2 = UnionInfo.id2
order by Jaccard desc;
```

2 SQL and Relational Algebra Queries

(a)

```
SELECT pid, store_id FROM
(SELECT pid, store_id, COUNT(*) as count_purchase FROM Purchases
GROUP BY pid, store_id) as Temp1
WHERE count_purchase = (SELECT MIN(Temp2.count_purchase) FROM
(SELECT pid, store_id, COUNT(*) as count_purchase
FROM Purchases GROUP BY pid, store_id) as Temp2
WHERE Temp1.pid = Temp2.pid);
```

(b)

```
SELECT P.cid, P.pid
FROM Purchases P
GROUP BY P.cid, P.pid
HAVING COUNT(P.pid) = 3;
```

(c)

```
Select DISTINCT P1.cid, P2.cid
FROM Purchases P1, Purchases P2
WHERE P1.cid < P2.cid AND
NOT EXISTS (Select P3.cid, P4.cid
FROM Purchases P4, Purchases P3
WHERE P3.pid = P4.pid AND P3.cid = P1.cid AND P4.cid = P2.cid);
```

(d)

```
(SELECT cid1, cid2 FROM
(SELECT P1.cid as cid1, P2.cid as cid2,P1.pid as pid
FROM Purchases P1, Purchases P2
WHERE P1.cid < P2.cid AND P1.pid = P2.pid) AS Temp
GROUP BY cid1 , cid2
HAVING (COUNT(DISTINCT(pid)) = (SELECT COUNT(DISTINCT(pid))
FROM Purchases P3 WHERE P3.cid = cid1))
AND (COUNT(DISTINCT(pid)) = (SELECT COUNT(DISTINCT(pid))
FROM Purchases P4 WHERE P4.cid = cid2)))
UNION
(SELECT C1.cid, C2.cid
FROM Customers C1, Customers C2
WHERE C1.cid < C2.cid AND
C1.cid NOT IN (SELECT cid from Purchases)
AND C2.cid NOT IN (SELECT cid from Purchases));
```

(e)

$$\begin{aligned}
&\rho(P1, \mathit{Purchase})\\
&\rho(P2, \mathit{Purchase})\\
&\rho(P3, \mathit{Purchase})\\
&\rho(P4, \mathit{Purchase})\\
&\rho(\mathit{AtLeastFour}, \pi_{P1.cid,P1.pid}(\sigma_{\theta_1}(P1 \times P2 \times P3 \times P4)))\\
&\rho(\mathit{AtLeastThree}, \pi_{P1.cid,P1.pid}(\sigma_{\theta_2}(P1 \times P2 \times P3)))\\
&\mathit{AtLeastThree} \setminus \mathit{AtLeastFour}
\end{aligned}$$

Where θ_1 is $P1.\mathit{cid} = P2.\mathit{cid} = P3.\mathit{cid} = P4.\mathit{cid} \wedge P1.\mathit{pid} = P2.\mathit{pid} = P3.\mathit{pid} = P4.\mathit{pid} \wedge (P1.\mathit{time} \neq P2.\mathit{time} \neq P3.\mathit{time} \neq P4.\mathit{time} \vee P1.\mathit{store\_id} \neq P2.\mathit{store\_id} \neq P3.\mathit{store\_id} \neq P4.\mathit{store\_id})$

Where θ_2 is $P1.\mathit{cid} = P2.\mathit{cid} = P3.\mathit{cid} \wedge P1.\mathit{pid} = P2.\mathit{pid} = P3.\mathit{pid} \wedge (P1.\mathit{time} \neq P2.\mathit{time} \neq P3.\mathit{time} \vee P1.\mathit{store\_id} \neq P2.\mathit{store\_id} \neq P3.\mathit{store\_id})$

(f)

$$\begin{aligned}
&\rho(C1, (\pi_{cid}\mathit{Customers}))\\
&\rho(C2, (\pi_{cid}\mathit{Customers}))\\
&\rho(\mathit{AllPairs}, \pi_{c1,c2}(\rho(C1.\mathit{cid} \to c1, C2.\mathit{cid} \to c2), \sigma_{C1.cid<C2.cid}(C1 \times C2)))\\
&\rho(P1, (\pi_{cid,pid}\mathit{Purchases}))\\
&\rho(P2, (\pi_{cid,pid}\mathit{Purchases}))\\
&\rho(\mathit{BadPairs}, \pi_{c1,c2}(\rho(C1.\mathit{cid} \to c1, P1.\mathit{pid} \to p1, C2.\mathit{cid} \to c2, P2.\mathit{pid} \to p2), \sigma_{\theta}(P1 \times P2)))\\
&\mathit{AllPairs} \setminus \mathit{BadPairs}
\end{aligned}$$

Where θ is $P1.\mathit{cid} < P2.\mathit{cid} \wedge P1.\mathit{pid} = p2.\mathit{pid}$

(g)

$$\rho(C1, (\pi_{cid} customers)$$

$$\rho(C2, (\pi_{cid} customers)$$

$$\rho(AllPairs, \pi_{c1,c2}(\rho(C1.cid \rightarrow c1, C2.cid \rightarrow c2), \sigma_{C1.cid<C2.cid}(C1 \times C2)))$$

$$\rho(Temp1, \pi_{c1,c2,Purchases.pid}(\sigma_{AllPairs.c1=Purchases.cid}(AllPairs \times Purchases)))$$

$$\rho(Temp2, \pi_{c1,c2,Purchases.pid}(\sigma_{AllPairs.c2=Purchases.cid}(AllPairs \times Purchases)))$$

$$\rho(T1, Temp1 \cup Temp2)$$

$$\rho(T2, Temp1 \cap Temp2)$$

$$\rho(BadPairs, \pi_{c1,c2}(T1 \setminus T2))$$

$$AllPairs \setminus BadPairs$$